AFRL-RH-WP-TR-2008-0090

UMCE-FM: Untethered Motion Capture Evaluation for Flightline Maintenance Support

Joseph T. Kider Jr.
Catherine R. Stocker
Norman I. Badler

Center for Human Modeling and Simulation
Department of Computer and Information Science
University of Pennsylvania
Philadelphia PA 19104-6389

June 2008

Final Report for April 2007 to June 2008

Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Readiness Research Division
Logistics Readiness Branch
Wright-Patterson AFB OH 45433-7604

NOTICE

AFRL-RH-WP-TR-2008-0090

//SIGNED//
ALEXANDER D. NELSON
Work Unit Manager
Logistics Readiness Branch

//SIGNED//
DANIEL R. WALKER, Colonel, USAF
Chief, Warfighter Readiness Research Division
Human Effectiveness Directorate
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
June 2008	Final	April 2007 – June 2008

4.TITLE AND SUBTITLE
UMCE-FM: Untethered Motion Capture Evaluation for Flightline Maintenance Support

5a. CONTRACT NUMBER FA8650-04-D-6546 DO07
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER 62202F

6. AUTHOR(S)
Joseph T. Kider Jr., Catherine R. Stocker, Norman I. Badler

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER 7184D208

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES)
Center for Human Modeling and Simulation
Department of Computer and Information Science
University of Pennsylvania
Philadelphia PA 19104-6389

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Materiel Command
Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Readiness Research Division
Logistics Readiness Branch
Wright-Patterson AFB OH 45433-7604

10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/RHAL

11. SPONSOR/MONITOR'S REPORT NUMBER(S) AFRL-RH-WP-TR-2008-0090

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
88th ABW/PA cleared on 02 October 2008, 88ABW-2008-0343.

14. ABSTRACT

The purpose of this research was to explore and evaluate the utility of novel motion capture technologies within the Air Force maintenance domain. The primary objective was to determine the potential of untethered motion capture capabilities for real-time human subject motion capture and performance data collection with full-scale physical props. The second objective was to evaluate data collected during maintenance task performance validation for the purpose of instruction generation and maintenance training. The effort consisted of domain analysis, conceptual design definition, prototype development, and a performance evaluation within relevant operational maintenance scenarios. Both university laboratories and field-based research environments were used to evaluate the efficacy of untethered motion capture.

15. SUBJECT TERMS Untethered Motion Capture, Air Force Maintenance, Flightline Maintenance, Job Performance Aids, Training

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Alexander D. Nelson
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR	40	**19b. TELEPHONE NUMBER** *(include area code)*

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. 239.18

THIS PAGE LEFT INTENTIONALLY BLANK

ABSTRACT

Aircraft maintenance is a core function in support of Air Force operations. The maintenance function encompasses tasks such as aircraft servicing, launch and recovery, scheduled maintenance, component repair, as well the technical training of new recruits. A variety of potential health and safety hazards exist in the environments where maintenance tasks are performed. Therefore, accomplishing maintenance tasks correctly is fundamental to personal safety and equipment integrity. The ability to accurately detect and recognize the actions of personnel performing maintenance tasks without the constant oversight of a maintenance "expert" or instructor would be advantageous to the training of new recruits. This project explored and evaluated the utility of using a contemporary full body motion capture suit including any software framework constraints associated with the insertion of this technology within the aircraft maintenance domain. A prototype motion capture recognition system determines a subject's current task from among a set of potential maintenance-like operations. This design includes the functionalities, constraints, and data requirements for motion-capture-based maintenance training aids.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. SUMMARY

The primary objective of this research project is to test the ease of use, reliability, and feasibility of untethered motion capture systems in the maintenance domain. A variety of sensors were tested to determine which worked best in situations comparable to a flightline maintenance environment. A second objective is to evaluate the motion data collected during a maintenance task scenario for the purpose of instruction or procedure generation and maintenance training.

In Section 2, we outline the general technological aspects of human motion capture that would allow field use (outside the laboratory setting) in typical Air Force maintenance environments. Section 2 also discusses and evaluates the current state-of-the-art in commercial, untethered motion capture technology. Section 3 presents an overview of the task recognition system we have developed to be used in conjunction with an appropriate motion capture system. Section 4 covers the results of experiments that demonstrate our system's validity as a recognition tool. Section 5 concludes with recommendations for future work on a virtual coaching system as an instructional aid.

2. INTRODUCTION

Maintenance tasks are fundamental operations in various training and repair environments, such as the flightline at an Air Force base, a hangar facility for scheduled maintenance, and at a technical training school for new recruits. Providing a training system that automatically detects and monitors a novice subject's actions has potential advantages. First, it provides the user the ability to experience a training environment containing 3D models of virtual equipment that do not deteriorate under the wear and tear of repetitive operations. Second, this system provides a safe environment for the subject to learn about various hazards without putting the trainee or the equipment at risk. Finally, it provides the basis for a training system in which all maintenance trainers can input their knowledge, creating a "super-trainer" system to monitor the activities of novices.

We thus explored and evaluated the utility of using a contemporary full body motion capture suit including any framework or architecture constraints associated with the insertion of this technology within the aircraft maintenance domain. Additionally, we developed a prototype motion capture architecture system to determine a subject's current task from among a set of potential maintenance-like operations. This design includes the functionalities, constraints, and data requirements for motion-capture-based maintenance training aids.

2.1 Evaluation of Motion Capture Technology

A *Motion Capture* system tracks human movements several times per second and records the data. This data can be used to animate virtual humans, monitor the behavior of the

real human in the motion capture suit, and do subsequent data analysis of the subject's performance. Available motion capture technology systems that are potentially suitable for the aircraft maintenance domain are limited for two main reasons. First, "tethered" hardware systems require the subject to perform actions in a predefined ("studio" or laboratory) capture space and not an arbitrary maintenance environment, and are thus unusable in a maintenance scenario in which the equipment cannot be transported into the small, predefined space. Secondly, the magnetometers used by "untethered" systems pose a potential problem in their interaction with the metal inherent to aircraft maintenance scenarios.

We investigated four commercial motion capture suits and systems that we believed might be appropriate for untethered use in cluttered metallic environments: the **Innalab 3DSuit, Animazoo GypsyGyro-18,** the **Xsens Moven,** and the **Measurand ShapeWrap II** (three of these suits are shown in Figure 1). All are based on inertial and gyroscope sensors or fiber optics to gauge movement and joint angles of the capture subject. We compared these four suits against a baseline optical motion capture system (our existing Ascension ReActor2 Digital Active-Optical Motion Capture System) and determined the Xsens Moven system was capable of overcoming most of the metallic interference in the environment. See Appendix A and Appendix B for more information on our tests.

Figure 1: Motion Capture Suits
(From left to right) Xsens Moven suit, Measurand ShapeWrap II, and the Animazoo GypsyGyro-18.

The Innalab 3DSuit captures movements via 17 inertial motion sensors placed on bands that wrap the performer's body. Each of the inertial motion sensors consist of gyroscopes, accelerometers, and magnetometers. The gyroscopes measure orientation based on the angular momentum of the limb they are tracking, the accelerometer measures the limb's acceleration, and the magnetometer measures position according to the surrounding magnetic field. Joint rotation is sensed around 3 axes and, with complex navigation algorithms, the sensors calculate a joint's pitch, yaw and roll angles. The communication device then combines the data from all sensors and calculates the positions of limbs relative to the "root bone", which is most often a foot that is in contact with the floor.

The Animazoo GypsyGyro-18 captures movements via 18 inertial rotational gyroscope sensors placed on the limbs on a spandex suit capturing up to 120 frames of motion per second. Each sensor has a published resolution of 0.01 degrees, and root mean squared (RMS) accuracy of 0.1 degree. The root position tracking is done by footstep calculating software. Orientations from the legs are extrapolated to give a position similar to the Innalab 3DSuit.

The Xsens Moven Inertial Motion Capture suit captures movements via 16 inertial rotational gyroscope sensors placed on the limbs on a spandex suit capturing up to 120 frames per second. The software factors simple acceleration into its sensors to allow short vertical changes such as jumping or cartwheel motions. The system integrates a biomechanical human model to more fully filter the input data for accuracy.

The Measurand ShapeWrap II suit uses 4 "ShapeTapes" that flex around the subject's arms and legs and senses data at 80-90 Hz. Additionally, the system uses 4 gyroscopes to alleviate "pose drift." Pose drift occurs when the captured joint data locations do not properly form a human skeleton, such as changing lengths of bone segments between joints. Pose drift primarily occurs because the system uses no fixed cameras or spatial reference points. Therefore, it is not capable of the high accuracy of camera-based systems. In particular, the ShapeWrap system cannot accurately establish the position of the person in a world coordinate system. The ShapeWrap II does not suffer from the occlusion problems that occur with optical systems when an object occludes the sensor and disrupts the capture. It also captures very smooth data regardless of position or orientation in the scene. The ShapeTape wires are extremely cumbersome, however, and could easily get snagged in a complex environment.

2.2 Testing Motion Capture in a Real World Environment

We tested the Xsens Moven suit with a maintenance person in a real environment (Figure 2) at the Naval Sea Systems Command (NAVSEA) base in Philadelphia. The NAVSEA operator dressed in the Moven suit and performed simple maintenance tasks on a test water filtration system. We captured and evaluated motion data from a series of tests to determine how accurate the system worked in a real scenario. Tests included walking about 100 meters away from the capture computer in the large warehouse space, replacing a cap on a water holder, opening a panel and screwing in a screw, climbing a

ladder, and doing a contrived reaching motion that required the subject to crawl around in the equipment.

Figure 2: Real World Environment Test
Images of Moven suit test at NAVSEA on real equipment.

All tests were successfully performed except for one. The ladder climbing test failed because the motion capture subject had to translate (move) vertically for long periods of time. This failure was expected because it is a built-in function of the Movens system to always consider one of the foot sensors to be in contact with the ground plane. The suit allows a subject's feet to be off the ground for only brief periods of time for actions such as walking, running, cartwheels, jumps, etc. When the subject crawled around in the equipment, 7 out of 8 captures returned good data, while one capture returned inaccurate data. In a cluttered warehouse, the subject walked a little over 100 meters before inaccurate data was returned. The screwdriver motions and water cap tests were performed accurately.

Hand poses and joint motions were independently collected using a right-hand wireless CyberGlove.

3. SYSTEM DESIGN METHODS

3.1 System Architectural Design

Figure 3 provides an overview of the general software system developed for this research project. The procedural components of the system are described below.

1. Subject selects a maintenance task: The subject specifies which task they will perform. This is a general task such as 'fix landing gear', or 'repair cockpit control panel'. This step ensures that the subject is performing a directed action,

and not randomly stringing motions together. Knowing the task allows the system to determine if anything is wrong (hazards or warnings) so it can prompt or correct the trainee.

2. Motion capture the task: The subject performs the task in a motion capture suit and their joint poses, angles, motion, and hand shapes are quantified and extracted.
3. Recognize the maintenance task: This is broken into two general parts. First, based on an empirical study of motion recognizers, we use “Motion Templates” to recognize the general category of the task the subject is performing. We augmented a list of binary relations to compress the motion capture data. Motion Templates suffer from differentiating actions that are numerically ‘close’. Secondly, we compute a simple distance measure to find the training instance closest to the given input motion to select the instanced class. This detects more detailed aspects of the action being performed.
4. Advance the finite state machine: This is a general overview of how the task is to be performed in a step-by-step manner as specified in Technical Order instructions or in their instantiation in our Parameterized Action Representation (PAR) system. This aids in action matching since we can break down a task into its relative parts, and recognize each part to ensure step correctness and that the task is completed in the correct order.
5. Finish the task: The task is completed.

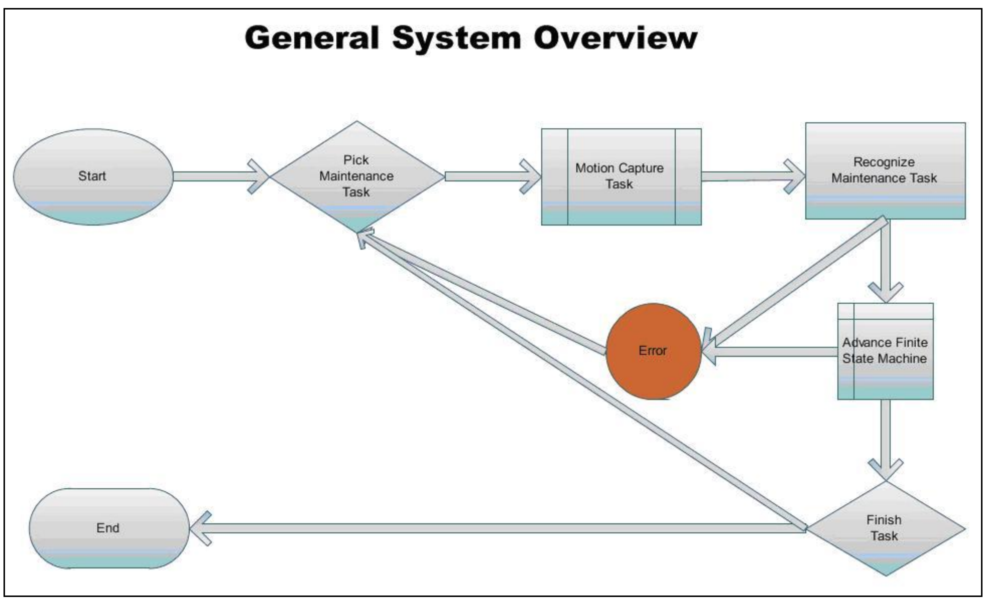


Figure 3: General Software System Overview

3.2 Demonstration Prototype Application

We designed a demonstration of the system described above with two objectives. The first objective was to test the suitability of the individual motion capture devices to detect tasks in a maintenance domain. The second objective was to evaluate the time and

accuracy performance of several possible algorithmic models used to classify the captured motions into discrete tasks. We used full body motion capture plus a CyberGlove for the right hand to collect and evaluate data. We focused on techniques that can organize, process, and navigate a database of motion capture clips of various maintenance motions. The subject is supplied with a database of maintenance ‘tasks’, which are simply state machines ordering motion-capture clips from our PAR database. This database of tasks is useful in organizing task information suitable for action matching, task analysis, and job aide information relative to a desired subject-to-system interactive maintenance behavior. The subject performs a ‘query’ sequence of motions and the system attempts to verify that the desired maintenance task is being completed accurately from the query sequence of motion clips being performed by the test subject.

To accomplish the second objective – evaluating the performance of various classification methods – requires first establishing the existing state of the art in motion capture technologies. Since real time performance would be essential to the training and safety of the repair technician, the faster the system can verify a correctly performed task the better. No classification technique is optimal for all classification tasks. By restricting the recognition system to a specific domain, however, a classifier can emerge that outperforms the others. Therefore an empirical survey is needed to determine a reasonable system.

3.3 Motion Clip Database

The action database consists of various motion clips captured from the motion capture system and CyberGlove data. We placed a grid of markers in space to simulate a work environment. We measured the 3D location of the markers so we could analyze the accuracy of our system. The motion classes we tested consisted of eleven short actions:

- Grasping the lower marker
- Grasping the middle marker
- Grasping the upper marker
- Using scissors on the lower marker
- Using scissors on the middle marker
- Using a screw driver on the lower marker
- Using a screw driver on the middle marker
- Tapping the upper cup with the palm of the hand
- Kicking randomly in the space
- Punching randomly in the space
- Throwing a ball to a fixed location from anywhere in the space

Though the actions described above may appear to be random, they do follow the Schlesinger-Schwartz grasping taxonomy. Heumer et al. showed that the Schlesinger-

Schwartz grasps were separable by performing a 2D visualization of the higher dimensional data, and thus this taxonomy was indeed separable and robust enough so that new actions could be added. Motions were hand-annotated with the type of motion that was performed at capture.

The longest motion in each of the 11 motion classes was uniformly time warped with each other so they would have the same temporal duration. Every motion in each class was dynamically time warped (DTW) to the longest motion in the class. Dynamic time warping warps two sequences non-linearly in the time dimension such that the minimum distance between two samples occurs when the signal features are aligned. For example, if two motions were performed at different frequencies, one faster than the other, DTW would align them properly so the temporal difference does not affect the classifier.

3.4 Task Database

The task database consists of small action state machines of ordered sequences of motion clips. For example: to open a drawer to a file cabinet, the subject first grasps the desired handle of the particular door, moves the thumb to unlatch the door, then pulls the arm back while grasping the handle. In our system, the task database consists of 5 simple sequences of actions so the open drawer task would consist of 5 states in the task's particular state machine: (1) grasping the drawer handle, (2) thumb unlatching, (3) arm movement start, (4) arm movement end, and (5) releasing the grasp on the drawer. A separate state machine is constructed ahead of time for each motion task based on the Parameterized Action Representation (PAR) system and the task's particular instructions. Our PAR system can be easily used to populate the task database from the task's particular instructions, breaking a task into smaller action state machines as necessary. While the subject is performing a requested task, the software system uses the PAR state machines to detect if the subject is not performing the correct task or taking too much time (stalled or confused) between tasks. Then a message can alert the subject that something is wrong.

4. RESULTS

4.1 Classifier Evaluation

In order to enable our system to correctly recognize and match a subject's input grasp or motion type as a corresponding grasp or motion class, various state-of-the-art techniques from pattern classification were applied and analyzed. Rather than focus on one particular method, a set of 28 different classifiers were tested to determine the most accurate and fastest for our test environment. We tested five types of classifiers: tree, lazy, function, probabilistic, and rule based classifiers:

(1) Tree classifiers (decision stumps, decision trees, and random trees) attempt to break up the classification task into a hierarchy of branch choices to lead to an end class.

(2) Lazy learners (Kstar, IB1, and LWL) search an existing database based on a distance metric to determine the closest class type.
(3) Function approximators (RBF, Perceptrons, and Regression) learn the parameters of a function from input data and return the output class.
(4) Probabilistic (Bayes) classifiers learn by building probability models from an example database.
(5) Rule induction methods (NNge, Jrip, and Conjunctive) create rules to determine the correct class.

Details of these particular methods are not important to this discussion, since ultimately performance accuracy and time are the crucial parameters.

The input ***training*** database consists of a table of pre-captured motion data tagged with the motion type for each ***example*** capture. The input ***test*** data consists of a single new motion capture take. The system then returns the class type of the input data. If classification of new motions is required by a task, data can be added to the training database to expand the system's recognition capabilities.

Each training example consists of frames of joint angles. A frame is captured at a rate of 60 samples a second and consists of 112 joint angles (J): 22 joint angle values from the CyberGlove, 90 joint angle values from our full body motion capture system (30 sensors $\times$ 3 dimensions). For each example motion there are F frames of data depending on the length of the captured motion. Therefore, each ***example*** motion in our database contains F $\times$ J feature dimensions. The input ***training*** database consisted of roughly 500 tagged ***example*** motions from 8 different subjects and tagged with the class type. The input ***test*** data consists of a new motion capture example with an unknown class.

First, we captured a large set of motions by various subjects to determine which classifier might be best suited for our system prototype for classifying motion capture data in a maintenance scenario. All results used 10-fold *cross validation* in the analysis. A 10-fold *cross validation* partitions the database into 10 random subset partitions: 33% used for testing data, 66% used for training data. The classifier is run on each of the 10 subset partitions and returns 10 values. These values are averaged for the final result given in our result tables. This approach guards against any bias in the data. We performed our tests in two different settings:

(1) All 8 subjects were used in both the training and testing data.
(2) A set of two random subjects were held out of the training data partitions.

This returns results for subjects who help train our system prototype, and subjects who do not train the system <u>and</u> have not been previously seen by our classifiers.

4.2 Static Grasp Type Recognition Results

First, to validate our general concept, we captured Schlesinger grasps using the CyberGlove from 8 different subjects. The Schlesinger grasp types included were

cylindrical, hook, lateral, palmar, spherical, and tip (Figure 4). A set of 28 different classifiers (see Table 1) were tested on the grasp dataset to determine the most accurate and the fastest classifier for our system prototype. In this case, the input ***training*** database consisted of a table of pre-captured grasps tagged with the grasp type for each ***example*** grasp. The input ***test*** data consisted of a new grasp capture. The system then returns the grasp type of the input data. Each ***example*** grasp contains a feature set of 22 joint angle values from the CyberGlove. The input ***training*** database consisted of roughly 1400 tagged ***example*** grasp types from 8 different subjects (22 joint angles × 1 frame for 1440 grasp examples).

Figure 4: Schlesinger Grasp Types

Images of the six Schlesinger grasp types we captured and tested. Source: Heumer et al., p. 3

Using 10-fold *cross validation*, the IB1 (nearest neighbor) classifier had a 98.45% correct classification and ran at 2.2 ms per grasp test. The IB1 classifier uses distance metrics to find the instance of training data that is closest to the given test instance, and predicts the same class as the closest example instance based on the distance metric. For grasps, our distance metric compares the 22 joint angles directly. Other classifiers were faster, such as a decision stump at 0.01 ms; however, the accuracy dropped to 32.36%. In this application, the IB1 classifier is the most accurate at a reasonable runtime rate that can easily work in real time. Therefore, our software system uses the IB1 classifier that correctly classifies grasp types in real time.

Table 1: Static Grasp Type Recognition Results

Summary of our results from the grasp classifier evaluation showing the IB1 method having the best classification percentage against 27 other classifiers showing a statistical analysis of various methods.

Classifier	Group	% Classified	Time/Test[ms]
IB1	lazy	98.45%	2.2
KStar	lazy	98.31%	29.62
MultilayerPerceptron	functions	96.97%	0.0607
RandomForest	trees	96.24%	0.03
LMT	trees	94.07%	0.09
SMO	functions	93.12%	0.07
Logistic	functions	92.97%	0.04

Classifier	Group	% Classified	Time/Test[ms]
SimpleLogistic	functions	92.50%	0.12
NNge	rules	92.01%	0.05
PART	rules	91.52%	0.03
NBTree	trees	91.18%	0.09
J48	trees	90.83%	0.03
RBFNetwork	functions	80.90%	0.08
JRip	rules	89.11%	0.12
REPTree	trees	88.13%	0.92
Ridor	rules	85.63%	0.34
BayesNet	bayes	83.71%	0.23
RandomTree	trees	82.22%	0.05
DecisionTable	rules	76.53%	0.04
NaiveBayes	bayes	76.32%	0.23
NaiveBayesUpdateable	bayes	76.32%	0.34
NaiveBayesSimple	bayes	73.88%	0.27
NaiveBayesMultinomial	bayes	72.09%	0.03
LWL	lazy	65%	21.3
ComplementNaiveBayes	bayes	63.67%	0.04
OneR	rules	43.89%	0.02
DecisionStump	trees	32.36%	0.01
ConjunctiveRule	rules	31.24%	0.02

4.3 Classifying Motion Data using Numerical Methods

The same systematic classifier evaluation (28 classifiers) was performed on the motion data in the motion clip database. In this evaluation, both full body motion capture and a CyberGlove were sampled for F frames of an example motion. Various motion classes were tested as previously noted: kicking, punching, screw driver motions at particular points, grasping objects at various points, using a scissor motion at a particular location, etc. Additionally, we further sub-sampled the motion data to just 10 frames per second (fps) and only used the top 80% of variance of joint angles with no loss of accuracy to improve speed (Tables 2 and 3).

Using 10-fold cross validation, the NNge method performed the best (97.4684% at 31ms per test). The NNge method is similar to the IB1 method; however it differs in that it uses non-nested, generalized exemplars in its distance metric search. Three classifiers: Multilayer Perceptron, Logistic Regression, and NBTree did not terminate after running more than one day making them obviously poor choices for a real time method. Further, for large datasets Multilayer Perceptron, Logistic Regression, and NBTree all ran out of memory.

The same systematic classifier evaluation was performed holding two random subjects out of the initial training data. Holding out subject data is a standard mechanism of testing and validating machine classifiers trained on a set of subject data acquisition runs to test how our system will perform with new subjects who were not used to train the system.

The numerical methods performed poorer only in cases where the held-out subject performed the actions differently than the subjects in the training data - primarily in the kicking and punching logical motions. Most errors occurred with classes such as kicking or punching since they were occasionally numerically similar to grasping a cup or motions performed at various frequencies (i.e. significantly faster or slower then training data), or different joint configurations (e.g. shoulder rotation could differ significantly but still accomplish a motion correctly). However, with numerical methods, the system performed more accurately for close actions. For example, if two screwdriver motions were very close together, the numerical methods could determine which motion was being performed. This test demonstrates that numerical methods help with the final location of the action but perform poorly if the action is initially performed in a unique way. For example, if two people perform a screwdriver motion slightly differently (unique shoulder rotation), numerical methods would classify each way as a distinct action rather than a screwdriver motion at a particular screw location, which is incorrect.

Table 2: Motion Data Results for All Subjects using Numerical Classifiers

Summary of our results from the motion data classifier evaluation using strictly numerical based methods, showing the NNge method has the best classification percentage against 27 other classifiers.

Classifier	Group	Correctly Classified (10 fold CV)	Time/Test[ms]
NNge	rules	97.4684%	31.01
RBFNetwork	functions	97.4684%	67.09
LMT	trees	96.2025%	3621.27
SimpleLogistic	functions	96.2025%	1781.14
SMO	functions	96.2025%	35.19
BayesNet	bayes	91.1392%	17.85
RandomForest	trees	89.8734%	41.77
IB1	lazy	88.6076%	1.65
PART	rules	78.4810%	52.66
J48	trees	77.2152%	19.24
DecisionTable	rules	72.1519%	145.57
Ridor	rules	72.1519%	128.10
NaiveBayes	bayes	67.0886%	3.54
NaiveBayesUpdateable	bayes	67.0886%	3.04
JRip	rules	63.2911%	47.47
LWL	lazy	55.6962%	1.65
RandomTree	trees	54.4304%	7.09
REPTree	trees	45.5696%	14.05
OneR	rules	43.0380%	2.03
NaiveBayesMultinomial	bayes	30.3797%	0.38
ComplementNaiveBayes	bayes	13.9241%	0.63
ConjunctiveRule	rules	11.3924%	14.05
KStar	lazy	8.8608%	0.13
DecisionStump	trees	5.0633%	5.70
MultilayerPerceptron	functions	N/A	N/A
Logistic	functions	N/A	N/A
NBTree	trees	N/A	N/A

Table 3: Motion Data Results for Unseen Subjects using Numerical Classifiers

Summary of our results from the Motion Data Classifier evaluation using strictly numerical-based methods showing the NNge method has the best classification percentage against the top classifiers. The second set of results holds out subjects and reruns the experiment showing how the system works for new subjects who have not trained the initial system; our classifiers still perform extremely well.

Classifier	Group	Correctly Classified (10 fold CV)	Time/Test[ms]
Tested on unseen subject			
NNge	rules	92.64%	31.01
RBFNetwork	functions	92.64%	67.09
LMT	trees	91.38%	3621.27
SimpleLogistic	functions	91.38%	1781.14
SMO	functions	91.38%	35.19
BayesNet	bayes	90.12%	17.85
RandomForest	trees	88.56%	41.77
IB1	lazy	85.67%	1.65
PART	rules	73.34%	52.66
All subjects in training			
NNge	rules	97.4684%	67.09
RBFNetwork	functions	97.4684%	31.01
LMT	trees	96.2025%	3621.27
SimpleLogistic	functions	96.2025%	1781.14
SMO	functions	96.2025%	35.19
BayesNet	bayes	91.1392%	17.85
RandomForest	trees	89.8734%	41.77
IB1	lazy	88.6076%	1.65

4.4 Classifying Motion Data using Logical Classification Methods

Müller and Röder described a method using motion templates (MT) to extract logically similar motions from a database rather than use numerical methods. First, a relational motion feature (RMF) matrix is constructed for each motion. This matrix describes the boolean geometric relations {0,1} between specified points of a pose. Applying a set of binary features (**bf**) relational motion features to a motion frame data stream of length F in a pose-wise fashion yields a feature matrix $X \in \{0,1\}^{(bf \times F)}$.

We constructed a set of 30 binary features to test our actions against, comprised of features from Müller and Röder, as well as some additional features that were relevant to the research (e.g., more x,y,z plane separation, acceleration features, and relative position features for the arms and leg sensors). These features are populated as the motion is captured via software. Table 4 provides examples of the binary relations tested, and Figure 5 shows a visualization of the binary features captured for some test motions.

Table 4: Examples of Binary Relations Tested

Left Hand Above Head?	Right elbow Bent (angle <120 degrees)?
Right Hand Above Head?	Left knee Bent (angle <120 degrees)?
Left Hand Above Waist?	Right knee Bent (angle <120 degrees)?
Right Hand Above Waist?	Left Hand moving away from body?
Left Foot in front of Body	Left Foot moving away from body?
Right Foot in front of Body	Right Foot moving away from body?
Shoulders Rotated Left (left forward)	Hand Approaching Each other?
Shoulders Rotated Right	Feet Approaching Each other?
Left elbow Bent (angle <120 degrees)?	Y-Extents Small?

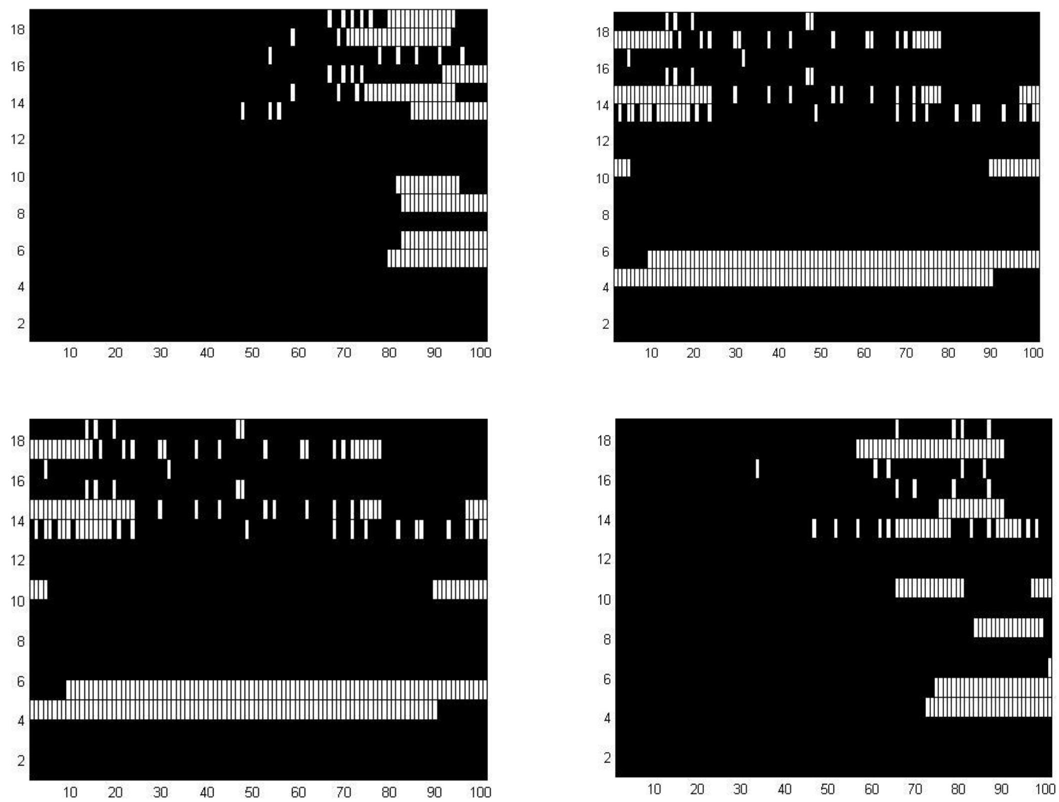


Figure 5: Relational Motion Feature Matrix Examples
Relational motion feature matrices for kick, lowercup, lowercup tap, and punch.

Motion templates (MT) are created by “learning” from examples in the training set of RMF matrices (see Figure 6). A single MT was created for each class of motions from all the classes’ RMF matrices. This is a real valued matrix between {0,1} that averages the RMF matrices. Each row of a MT corresponds to one relational feature similar to the RMF matrices. Given a set of N example motions for a specific motion class, we learn a meaningful MT that grasps the essence of a class.

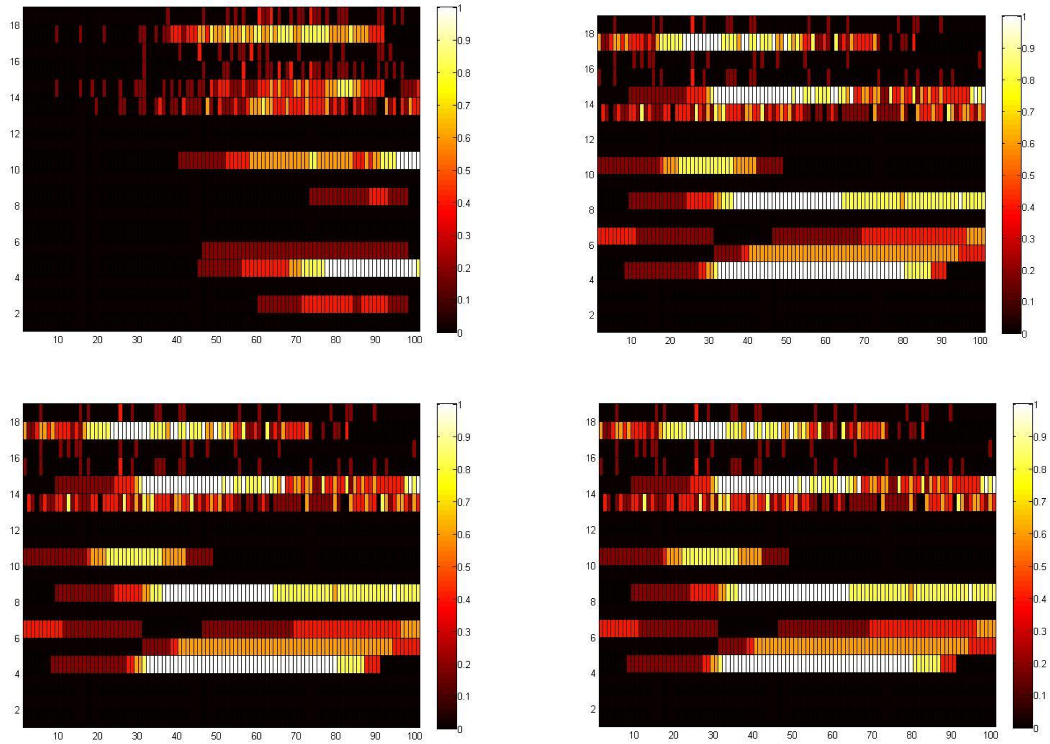


Figure 6: Motion Template Examples
Motion Template of lowercup tap, punch, kick, lowercup

4.5 Logical Methods Results

Motion Templates worked extremely well to differentiate general classes of motions that were very separable, for example, a kick against a screwdriver motion. Table 5 demonstrates this fact. Further analyzing the cases MT's failed we found that MTs were not very accurate when attempting to differentiate motions that are functionally close together. For example, if a person performs a screwdriver motion on one screw, then performs a screwdriver motion on a screw that is a couple of inches away from the initial screw; logical classifiers confuse which motion is being performed. We illustrate this confusion in Table 6. There are two possible solutions. The first one could resolve this problem with a better set of binary relation features, however breaking up the entire space will balloon the size of the feature matrices making the method intractable. The second method we propose is to construct a hybrid system that can be constructed which combines both logical methods and numerical methods described below.

Table 5: Motion Template Recognition Results

Classifier	**% correct**	**Time/Test[ms]**
Tested unseen subject		
MOTION TEMPLETES	94.43%	15.82
NNge	92.64%	67.09
RBFNetwork	92.64%	31.01
LMT	91.38%	3621.27

Classifier	% correct	Time/Test[ms]
All subjects trained		
MOTION TEMPLETES	98.6000%	15.82
NNge	97.4684%	67.09
RBFNetwork	97.4684%	31.01
LMT	96.2025%	3621.27

Table 6: Logical Motion Confusion Matrix

Summary of our results from the motion data classifier evaluation using strictly numerical based methods illustrating the confusion of logical classifiers. For motions that are very close, logical methods confuse which finer grain method is actually being performed, however they always return a general broad class correctly.

a	b	c	d	e	f	g	h	i	j	k	<-- classified as
21	0	1	1	0	0	0	0	0	0	0	a = lowercup
0	22	0	0	2	0	0	0	0	0	0	b = middlecup
2	0	10	0	6	0	0	0	2	0	0	c = punch
0	0	0	21	0	0	0	0	0	0	0	d = slowercup
0	0	0	0	21	0	0	0	0	0	0	e = smiddlecup
0	0	0	0	0	24	0	0	0	0	0	f = tlowercup
0	0	0	0	0	0	22	3	0	0	0	g = tuppercup
0	0	0	0	0	0	0	24	0	0	0	h = uppercup
0	0	0	0	4	0	5	0	19	0	2	i = kick
0	0	0	0	0	0	0	0	0	24	0	j = hlowerup
0	0	0	0	0	0	0	0	0	0	20	k = ball

Logically Different Motions
Motions of a Different Actor/ Rotations

4.6 Hybrid Logical/Numerical Classification Method

The final prototype was designed around a hybrid classification method to determine which task the subject performs. The test results for the hybrid classification method are provided in Table 7. First, a logical classifier is run to determine a broad range of what motion is being performed, such as a "right quadrant screwdriver motion." This broad class contains a hierarchy of finer grain motions which actually determine the final placement of the motion. To differentiate these, the results of the logical classifier were input to a numerical classifier for the sole purpose of operating on the set of motions in the logical motion classifier's output. This method greatly improves the cases where both methods fall short and gives our prototype more robustness and accuracy.

Table 7: Hybrid Recognition Results

Classifier	% Correct	Time/Test[ms]
Tested unseen subject		
Hybrid Method	99.93%	18.02

All subjects trained		
Hybrid Method	100%	18.02

4.7 Development Issues

During the implementation of our demonstration application we encountered a few issues that needed to be addressed. No shape-tape or inertial-based motion capture suit system performed as well as an optical motion capture system (such as our ReActor2), nor was this unexpected. The Measurand ShapeWrap II system performed extremely poorly in our initial evaluation at SIGGRAPH 2007, encountering a large amount of pose drift. The Innalab 3DSuit did not provide a software development kit (SDK) making use of the hardware for any real-time application impossible. The Animazoo GypsyGyro-18 required a tedious and laborious calibration step, and still did not perform well even in our lab with simple metals nearby such as a file cabinet or steel in the concrete floor. However, after rigorous evaluation we determined the Xsens Moven Inertial Motion Capture system performed the best and most robustly under our tests. Still the system suffers from magnetic interference when in direct contact with hard metals, such as a heavy iron, but direct contact is a rare scenario.

Overall, the CyberGlove is a well-designed and reliable device. The licensing of the SDK, however, caused a few problems. When installing the SDK, a code is generated. This code is then emailed to Immersion which returns another code to be entered in the authorization software to permanently unlock or authorize the software. In itself, this is not a bad procedure; however, this procedure only authorizes the software for one user on the computer where it is installed. Installing the software on another computer or reinstalling the software on the same computer or allowing another user on the computer to use the software, requires sending and receiving a new code from Immersion.

Motion capture, in general, is a laborious process. Every capture session requires the subject to wear cumbersome sensors, undergo some calibration procedure, and stop and start motions to get clean data. An interesting direction of future work would be to conduct a statistical analysis on how many sensors are actually required to get accurate results. For example, if we start removing sensors, reducing their number from 16 to 1, can we still recover the motion the subject is performing accurately? How less intrusive can we make motion capture in general so it can be seamlessly integrated into flightline training maintenance scenarios? These questions may have to wait for better commercial motion capture technologies.

5. CONCLUSIONS AND RECOMMENDATIONS

Throughout this project, we have been concerned with testing the ease of use, reliability and feasibility of untethered motion capture systems in the maintenance domain. Although untethered systems are necessary to enable the movement of maintenance technicians, our main concern has been the reaction of the magnetometers used by these systems to the metal inherent in aircraft maintenance environments. After comparing three different untethered suits to our own (“tethered”) optical motion capture system –

which is highly accurate – we have determined that the Xsens Moven motion capture suit is the most reliable untethered system and suitable for those scenarios. For this reason, we recommend the Xsens system to be the most suitable to monitor the motions of people involved in maintenance tasks during some phases of their training.

In line with this recommendation, we have taken the first step towards a training system that uses motion capture. Our system has learned to recognize a series of actions and tested well on people of varying height, weight and gender. In the future, the results from this system can be fed back into a "virtual coaching system" to monitor a trainee's actions and warn or correct off-nominal performance in real time. Work has begun on the next phase of this effort to investigate the addition to Air Force training materials of an interactive, computer-generated human agent to guide both a subject and a team of subjects during a maintenance task training scenario and to measure, evaluate, and verify the trainee's actions against an established model of task performance.

6. REFERENCES

6.1 History of Motion Capture

[1] Badler, N., Hollick, M. J., & Granieri, J. P. Real-time control of a virtual human using minimal sensors. *Presence: Teleoperators and Virtual Environments*, 2(1), 1993, pp. 82-86.

[2] Calvert, T. W., Chapman, J., & Patla, A. Aspects of the kinematic simulation of human movement. *IEEE Computer Graphics and Applications,* 2(9), 1982, pp. 41-50.

[3] Kines, M. Planning and directing motion capture for games. 2000. http://www.gamasutra.com/features/20000119/kines_01.htm.

[4] Kleiser, J. *Character motion systems*. Course Notes: Character Motion Systems, ACM SIGGRAPH, Anaheim, CA, August 1993, pp. 33-36.

[5] Maiocchi, R. 3-D character animation using motion capture. In N. Magnenat-Thalmann & D. Thalmann (Eds.), *Interactive Computer Animation*. Prentice-Hall, 1996, pp. 10-39.

[6] Menache, A. *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann Publishers Inc., 1999.

[7] Sturman, D.J. A *brief history of motion capture for computer character animation*. http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/motion_capture/history1.htm

[8] Trager, W. A practical approach to motion capture: Acclaim's optical motion *capture system*. Character Motion Systems. SIGGRAPH Course 9, 1994. http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/motion_capture/motion_optical.htm>

6.2 Survey of Motion Capture Systems

[9] Animazoo - Motion Capture Systems. http://www.animazoo.com/capturedSystems.aspx

[10] Ascension Products – ReActor. http://www.ascension-tech.com/products/reactor.php

[11] Measurand ShapeWrap II Overview. http://www.measurand.com/products/ShapeWrap_overview.html

[12] Xsens, Overview of Moven Inertial Motion Capture Technology, http://www.xsens.com

6.3 Motion Recognition

[13] Arikan, O., Forsyth, D. A., & O'Brien, J. F. Motion synthesis from annotations. *ACM TOG,* 22(3), 2003, pp. 402-408.

[14] Basu, S., Shanbhag, S., & Chandran, S. Search and transitioning for motion captured sequences. In *VRST '05, Proceedings of the ACM Symposium on Virtual* Reality *Software and Technology*, 2005, pp. 220-223.

[15] Brand, M. & Hertzmann, A. Style machines. In *Proceedings of the 27th Annual ACM SIGGRAPH Conference on Computer Graphics and Interactive Techniques,* 2000, pp. 183-192.

[16] Green, R. & Guan, L. Quantifying and recognizing human movement patterns from monocular video images. *IEEE Transactions on Circuits and Systems for Video Technology, Part I,* 14(2), 2004, pp. 179-190.

[17] Heumer, G., Amor, H.B., Weber, M., & Jung, B. Grasp recognition with uncalibrated data gloves - A comparison of classification methods. In *VR '07, IEEE Virtual Reality Conference*, 2007, pp. 19-26.

[18] Liu, G., Zhang, J., Wang, W., & McMillan L. A system for analyzing and indexing human-motion databases. In *SIGMOD '05, Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data,* 2005, pp. 924-926.

[19] Liu, G. & McMillan, L. Segment-based human motion compression. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2006, pp. 127-135.

[20] Müller, M. & Röder, T. Motion templates for automatic classification and retrieval of motion capture data. In *Proceedings of the 2006 ACM SIGGRAPH/ Eurographics Symposium on Computer Animation,* 2006, pp. 137-146.

[21] Müller, M., Röder, T., & Clausen, M. Efficient indexing and retrieval of motion capture data based on adaptive segmentation. In *CBMI 2005, Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing,* 2005, Riga, Latvia.

[22] Müller, M., Röder, T., & Clausen, M. Efficient content-based retrieval of motion capture data. *ACM Transactions on Graphics,* 24(3), 2005, pp. 677-685.

[23] Rabiner, L. R. & Juang, B. H. *Fundamentals of Speech Recognition*. Prentice Hall: Signal Processing Series, 1993.

[24] Rose, C., Cohen, M. F., & Bodenheimer, B. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5), 1998, pp. 32-40.

[25] Schlesinger, G. Der mechanische aufbau der künstlichen glieder. In M. Borchardt et al., (Eds), *Ersatzglieder und Arbeitshilfen für Kriegsbeschädigte und Unfallverletzte*, Berlin: Springer-Verlag, 1919, pp. 321-699

[26] So, C.K.F. & Baciu, G. Entropy-based motion extraction for motion capture animation: Motion capture and retrieval. *Computer Animation and Virtual Worlds,* 16(3-4), 2005, pp. 225-235.

[27] Taylor, C.L. & Schwarz, R.J. The anatomy and mechanics of the human hand. *Artificial Limbs*, 2, 1955, pp. 22-35.

7. LIST OF ABBREVIATIONS

DTW:	Dynamically Time Warped
EDR:	Enhanced Data Rate
LED:	Light-Emitting Diode
MT:	Motion Templates
NAVSEA:	Naval Sea Systems Command
PAR:	Paramaterized Action Representation
RMF:	Relational Motion Feature
RMS:	Root Mean Square
SDK:	Software Development Kit
USB:	Universal Serial Bus

APPENDIX A – DETAILED HARDWARE INFORMATION

For the purpose of our evaluation of motion capture systems in maintenance domain environments we tested an **Immersion Wireless CyberGlove II** and three motion capture suits the **Innalab 3DSuit, Animazoo GypsyGyro-18,** and the **Xsens Moven**. We compared these three suits against a baseline optical motion capture system: **Ascension's ReActor2: Digital Active-Optical Motion Capture System**.

Immersion Wireless CyberGlove II

This new wireless CyberGlove II (Figure A1) system provides 22 high-accuracy joint-angle measurements. It uses resistive bend-sensing technology to transform hand and finger motions into real-time digital joint-angle data. Each sensor is extremely thin and flexible and is virtually undetectable in the lightweight elastic glove. The basic CyberGlove II system includes one data glove, two batteries, a battery charger, and a universal serial bus (USB) Bluetooth technology adapter with drivers. The CyberGlove II has 0.5° sensor resolution and sensor repeatability (average standard deviation between glove donnings) of 1°. The typical data rate is 100 records per second. Its operating range is within a 30 foot radius of the USB Bluetooth adapter.

Figure A1: Immersion Wireless CyberGlove II

General Motion Capture Suit Specifications

For our purposes we will examine five popular motion capture systems in detail:

a) A tethered 'capture studio' setup: Ascension's ReActor2: Digital Active-Optical Motion Capture System
b) A portable system: Innalab 3DSuit Inertial Motion Capture
c) A portable system: Animazoo GypsyGyro-18 Inertial Motion Capture
d) A portable system: Xsens Moven Inertial Motion Capture
e) A portable system: Measurand ShapeWrap II Inertial

Ascension's ReActor2: Digital Active-Optical Motion Capture System

The ReActor2 captures movements via 30 active optical markers placed carefully on the joints of the performer's bodysuit. Each optical marker consists of 42 embedded infra-red

LEDs flashing at 900Hz, which translates to about 30 frames of 30 marker animation data per second.

ReActor2 capturing takes place in a fixed framed motion capture area. The area consists of 8 bars containing 544 digital detectors that actively track the signals from the markers when in the 3m x 4.2m x 2.4m rectangular space. Data is collected by these detectors and sent to a PC at a rate of 900 measurements per second.

Captured data is processed by Ascension's FusionCore software for real-time visualization, editing, and exported via Ethernet to applications such as Alias's Motion Builder.

Innalab 3DSuit Inertial Motion Capture System

The Innalab 3DSuit captures movements via 17 inertial motion sensors placed on bands that wrap the performer's body. Each sensor operates at 120Hz, has a resolution of 0.01 degrees, and RMS accuracy of 0.7 degree in yaw, 0.1 degree in pitch and roll. Each of the inertial motion sensors consist of gyroscopes, accelerometers and magnetometers. The gyroscopes measure orientation based on the angular momentum of the limb they're tracking, the accelerometer measures the limb's acceleration and the magnetometer measures position according to the surrounding magnetic field. Joint rotation is sensed around 3 axes and, with complex navigation algorithms, the sensors calculate a joint's pitch, yaw and roll angles. The communication device then combines the data from all sensors and calculates the positions of limbs relative to the "root bone," which is most often a foot that is in contact with the floor. The 3DSuit does not require the performer to reside in a capture area, but instead uses a Bluetooth 2.0 enhanced data rate (EDR) class 1 USB 2.0 (VCP Baud Rate 800Kb) interface which requires the performer to be located within 100m of the PC capturing the data.

Captured data is processed via 3DSuit software and then transferred to Alias' Motion Builder. Similar to the ReActor2 all these steps are performed "on-the-fly" with minimum time lag.

Animazoo GypsyGyro-18

The Animazoo GypsyGyro-18 captures movements via 18 inertial rotational gyroscope sensors placed on the limbs on a spandex suit capturing up to 120 frames of motion per second. Each sensor has a published resolution of 0.01 degrees, and root mean squared (RMS) accuracy of 0.1 degree. The root position tracking is done by footstep calculating software. Orientations from the legs are extrapolated to give a position similar to the 3DSuit. Also, similar to the 3DSuit, the GypsyGyro-18 is not confined to a motion capture area however; it must be used within 100m of the computer capturing the data.

Captured data is processed via Animazoo software and then transferred to Alias Motion Builder. Similar to the other two suits these steps are performed with minimum lag.

Xsens Moven Inertial Motion Capture

The Xsens Moven Inertial Motion Capture suit captures movements via 16 inertial rotational gyroscope sensors placed on the limbs on a spandex suit capturing up to 120 frames of motion per second. The software factors simple acceleration into its sensors to allow short vertical changes such has jumping or cartwheel motions. The system integrates a biomechanical human model to more fully filter the input data for accuracy.

The Moven is not confined to a motion capture area however; it must be used within 100m – 300m of the computer capturing the data.

Measurand ShapeWrap II

The Measurand ShapeWrap II uses 4 ShapeTapes that flex around the subject's arms and legs and runs at 80-90 Hz. Additionally the system uses 4 gyroscopes to alleviate pose drift.

Since the system uses no fixed cameras or fields, it is not capable of the high accuracy of camera-based systems, particularly for position of the person in a world coordinate system. But, ShapeWrap II does not suffer from occlusion, and captures very smooth data regardless of position or orientation in the scene. Additionally, the shapetape wires are extremely cumbersome and could easily get snagged in a complex environment.

APPENDIX B – DETAILED MOTION CAPTURE SYSTEM TESTING

Subjective Performance Analysis

We rigorously tested each motion capture suit (except the Measurand ShapeWrap II suit which failed preliminary tests) and report on the results of each test.

Setting up the Suit

The ReActor2 requires the subject to wear a spandex suit with Velcro patches, to allow for proper marker placement. The suit becomes hot during extended use and it also requires that a performer be of a very specific size and height. Setup took our team around 40-45 minutes. Each marker must be re-placed on the suit every time the subject performs in it. Furthermore, the ReActor2 requires a tedious setup placing the wires from the markers into the correct order on the control belt before every usage.

The Innalab 3DSuit consists of a series of bands that are placed over work clothes, plus pads to be positioned in the performer’s shoes. It is very comfortable for the performer and the setup took only about 5-10 minutes.

The Animazoo GypsyGyro-18 and Xsens Moven requires the subject to wear a spandex suit with sensors built in. The suit set up took only about 5-10 minutes to dress the suit. The Animazoo suit required an initial calibration stage before usage that required the subject to hold a T-pose for around 3 minutes which was highly uncomfortable for the subject. The Xsens suit did not require this tedious setup process.

Setting up Motion Builder

The 3DSuit requires a plug-in to Motion Builder, but required only simple scaling and had an intuitive interface. Most of the calibration was completed after the actor stood in a “T-pose” while one button was pressed. This setup took only a couple minutes.

The ReActor2 requires the software user to import the marker data from a plug-in. Upon receiving the marker positions, the actor must be scaled and oriented to the dimensions of the character’s markers (it should be noted that many in our team have experience with this setup and still no one ever gets the scaling and orientation fully accurate since the setup is so tedious). Then each marker must be dragged to the appropriate place on the model’s actor. Total time to setup took about 10 minutes.

The GypsyGyro-18 and Moven processes the raw data via internal software then outputs BVH (Biovision hierarchical) data directly. This data can be used in commercial animation software such as Motionbuilder and Maya.

Reaching for Objects

The subject was prompted to reach for each particular point on the grid (Figure B1). The locations in space were measured by hand, and then compared to the data from all suits tested to determine accuracy. All systems produce appropriate reaching behaviors, but the objective reaching tests of the systems – which measure the scale of the reach – are discussed in the numerical results section.

Figure B1: Reaching for Objects Example
The left avatar is the 3DSuit, right is the ReActor2.

Climbing a Ladder

The subject was required to climb a ladder and perform similar reaching motions as discussed in the previous test. The 3DSuit, GypsyGyro-18, Moven hardware does not allow vertical translation of the character, so as a person climbs a ladder, the corresponding character makes climbing motions, but ultimately stays grounded. The ReActor2 translates upward in space appropriately. The Figure B2 shows the example of an inertial based system failing to capture the appropriate vertical translation.

Figure B2: Climbing a Ladder Test
The left avatar is the 3DSuit, right is the ReActor2

'Occluded' Reach

The subject was required to lift an object from the bottom of a trash can. This test illustrates the first problem with optical systems, namely occlusion. Inertial systems do not suffer this problem. The 3DSuit, GypsyGyro-18, and Moven are able to capture reaching motion in a small, enclosed area, while the ReActor2 is not. This is because the detectors in the ReActor2 cannot determine the location of a marker if the light-emitting diodes (LED's) in the marker are occluded. The inertial suits do not suffer from this problem. Figure B3 illustrates this issue.

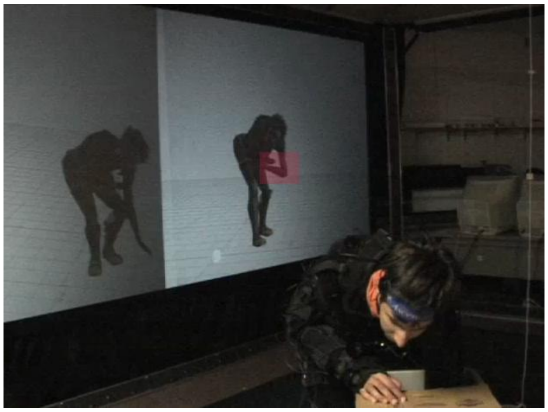


Figure B3: Occluded Reach Test
The left avatar is the 3DSuit, right is the ReActor2

Self-Reference Test

Here the subject was required to perform self-reference tests (Figure B4). The subject first touched both hands together away from the body making sure both sensors were touching, and then touched the sensors on the upper body and legs with the hand sensors. Since we know what sensors are being touched, one way to determine accuracy is to check whether the hand sensor and the sensor it touched had similar positions (accounting for the size of the sensor itself). The 3DSuit appeared to perform better during the Self-Referencing task than the ReActor2, the objective referencing tests of each system are discussed below. The Moven performed the best out of the 3 inertial suits tested for this particular scenario.

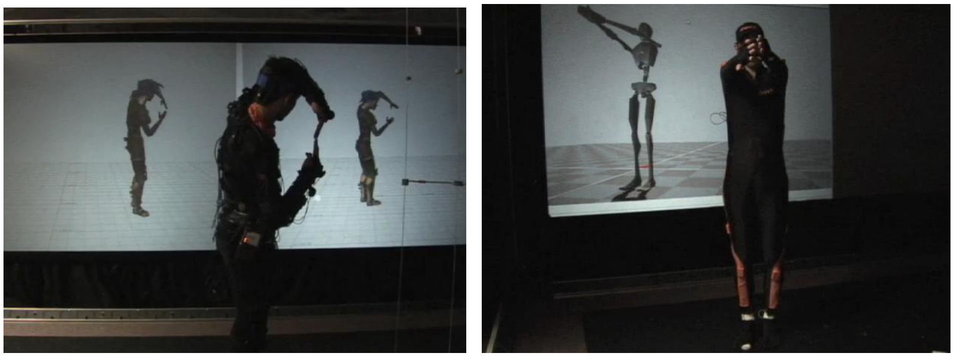


Figure B4: Self-Reference Test
The left avatar is the 3DSuit; middle is the ReActor2; far right is the Moven.

Exiting the 'Capture Studio'

The subject was prompted to leave the capture volume (Figure B5). This test simply illustrates the second disadvantage of optical systems: they are confined to a predefined enclosed space. The 3DSuit, GypsyGyro-18, and Moven suit are capable of exiting the 'capture studio' and we even found that we could capture the subject while he was on the other side of a brick wall, in a closed room. The ReActor2 is confined to the 'capture studio' with the required detectors.

Figure B5: Exiting the 'Capture Studio'
The left avatar is the 3DSuit, right is the ReActor2.

Simple Metal Experiment Capture

In this test the subject performed the self reach test but placed some of the sensors on a metal file cabinet (Figure B6). This test demonstrated a simple metallic environment. We see the Animazoo GypsyGyro-18 (right image) arms crossed severely instead of touching showing huge pose drift where the Xsens Moven suit did not. However, the red circles indicate that there was some metallic interference.

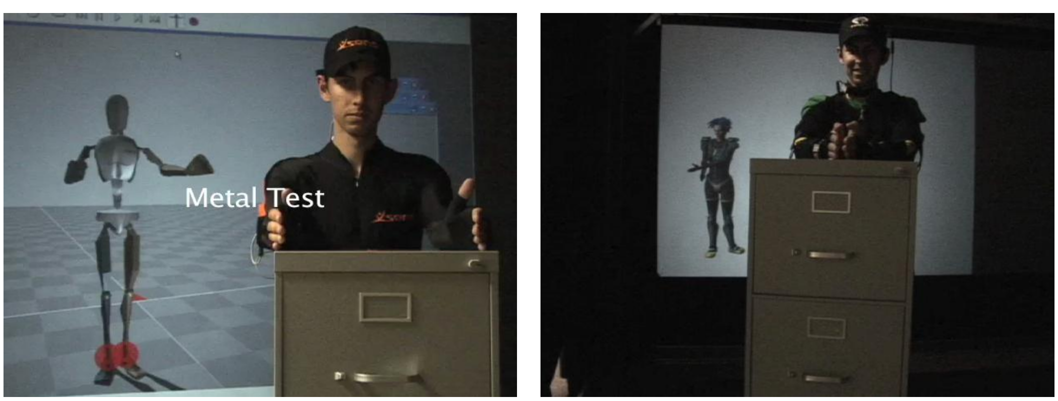


Figure B6: Simple Metal Experiment Capture
The left avatar is the Moven; the right image is the GypsyGyro-18.

Walking Drift Test

This test required the subject to walk around the environment and return to the same location marked with a piece of tape placed on the floor. Our hope was to confirm the start and end locations were the same. This did not happen with the inertial systems. They returned a slight error after a prolonged capture since they do not capture global positioning well. The Moven (Figure B7) had the best results. The GypsyGyro-18 (Figure B8) drifts more than the Moven avatar during captures sessions.

This test demonstrates that before this system is used in a real environment, the subject should conduct an initial orientation step to confirm their exact location relative to the equipment or task.

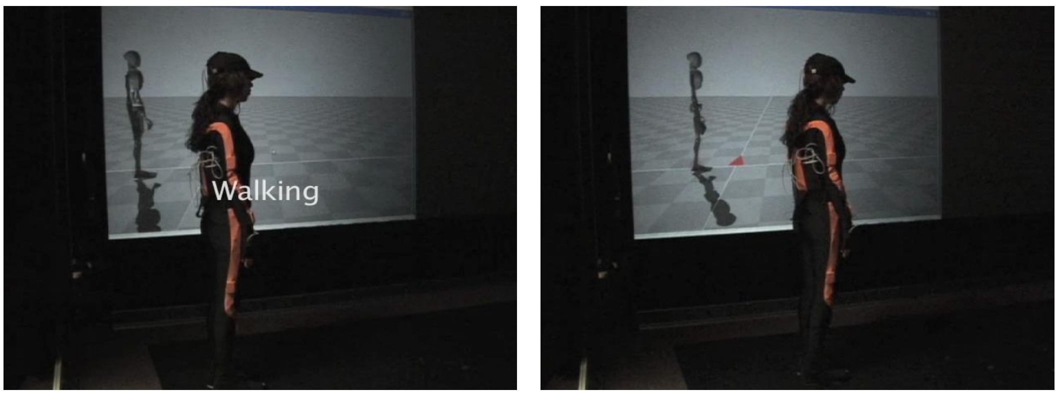


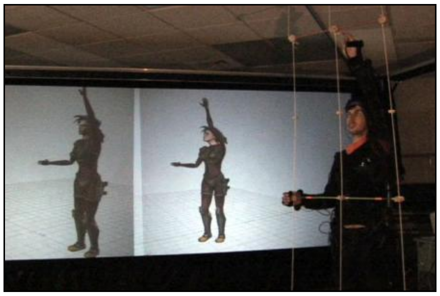
Figure B7: Walking Drift Test (Moven)
Left image is prior to walking. Right image is after the subject returned to the same physical location.

Figure B8: Walking Drift Test (GypsyGyro-18)
Left image is prior to walking. Right image is after the subject returned to the same physical location.

Objective Performance Analysis

We attempted to concretely analyze the performance of each suit in controlled laboratory tests to gauge the accuracy of the data. The performer wore both the 3DSuit and the ReActor2 simultaneously while performing a variety of motions multiple times (Figure B9). We then averaged the L2 error at each point and presented those numbers. We then

averaged the general L2 error for each suit per test. The Moven and GypsyGyro-18 suits were tested on two separate occasions. However, both suits were not worn simultaneously since we already had the baseline data from the ReActor2 and two suits at once was very uncomfortable for the test subject.



Reaching Points Test

Self-Reference Test

Figure B9: Objective Performance Examples

Reaching to Set Marker Test

Here the performer touched various marker points in 3D space on a grid hung from the ceiling in our motion capture lab. The points in the space were measured with a tape measure to get their actual coordinate and then were compared to the data received from the two suits. The grid was 3×3 rectangle. No inertial system performed as well as an optical system; however the Xsens Moven system showed the least error (Table B1).

Table B1: Motion Capture Suit Marker Position Test Errors

Average error for the grid of marker points. The marker position in 3D space was hand measured and the end vector location was analyzed using the Motion builder software to determine the end vector's position. Tests were performed several times and the errors were averaged out for each marker position in the table.

Hand	**Marker Position**	**Avg. ReActor2 Error**	**Avg. 3DSuit Error**	**Avg. GypsyGyro-18 Error**	**Avg. Moven Error**
Right	Lower Left	70.5 mm	109.6 mm	89.3 mm	85.2 mm
Right	Middle Left	39.7 mm	110.6 mm	122.2 mm	76.2 mm
Right	Upper Left	2.2 mm	171.1 mm	141.5 mm	151.3 mm
Right	Lower Center	85.4 mm	83.7 mm	89.6 mm	73.8 mm
Right	Middle Center	7.6 mm	60.1 mm	56.2 mm	53.2 mm
Right	Upper Center	24 mm	144.1 mm	141.5 mm	94.3 mm
Right	Lower Right	1.4 mm	53.1 mm	51.8 mm	50.1 mm
Right	Middle Right	18.2 mm	61.8 mm	67.1 mm	51.9 mm
Right	Upper Right	11.5 mm	137.9 mm	117.0 mm	87.6 mm
Left	Lower Left	11.4 mm	41.2 mm	43.6 mm	32.5 mm
Left	Middle Left	7 mm	25.7 mm	25.7 mm	15.8 mm
Left	Upper Left	33.5 mm	136.25 mm	116.1 mm	96.5 mm
Left	Lower Center	25.6 mm	59.1 mm	54.7 mm	39.4 mm
Left	Middle Center	15.6 mm	60.1 mm	61.8 mm	20.1 mm
Left	Upper Center	28.9 mm	72.6 mm	78.2 mm	70.6 mm

Hand	Marker Position	Avg. ReActor2 Error	Avg. 3DSuit Error	Avg. GypsyGyro-18 Error	Avg. Moven Error
Left	Lower Right	62.4 mm	62 mm	62.8 mm	34.2 mm
Left	Middle Right	16.4 mm	59.7 mm	51.1 mm	51.7 mm
Left	Upper Right	38.6 mm	80.2 mm	81.4 mm	60.4 mm
AVG ERROR:	**------------------**	**27.8 mm**	**84.8 mm**	**80.64 mm**	**63.6 mm**

Self-Reference Test

Here the performer touched (as close as possible) one marker with another to asses a 3D positional error. Eight points were recorded and tracked (Table B2).

Table B2: Motion Capture Self-Reference Test Errors

Average error for self-reference tests. The two marker positions were analyzed in motion builder and the error reports the average of the difference in recorded position between them over several trials.

Test Markers	Avg. ReActor2 L2 Error	Avg. 3DSuit L2 Error
Right Hand to Right Head	78.2 mm	89.2 mm
Left Hand to left Head	76.7 mm	84.5 mm
Right Hand to Right Knee	80.8 mm	129.1 mm
Left Hand to Left Knee	65.6 mm	83.4 mm
Right Hand to Right Toe	67.2 mm	144.6 mm
Left Hand to Left Toe	41.4 mm	104.1 mm
Left Hand to Left Ankle	48.2 mm	48.5 mm
AVERAGE L2 ERROR:	**65.5 mm**	**97.6 mm**

Note: It should be noted that the markers for each suit do not exactly correspond with the other suits. This is obvious since the ReActor2 has 30 markers and the Innalab 3DSuit has 17. We took a measurement from the Innalab 3DSuit to the marker position on the ReActor2 and adjusted the data afterwards. This is apparent with the head, since the highest Innalab sensor only reaches the base of the neck, we took a measurement from that sensor to the position the ReActor2's forward head markers were (point of head self reference) and subtracted the error to make the test more fair in the resultant matrix.

Motion Capture Suit Advantages/Disadvantages

Ascension ReActor2

Advantages

- Fewer occlusions from cameras in passive systems employing 6-24 cameras.
- No metallic distortions from surrounding environment.
- Accurate capturing of a performer's vertical translation position.
- Accurate capturing of running, jumping, and motions with both feet off the ground.
- An older system. There is better support and documentation.

Disadvantages

- Subject’s comfort level and setup of equipment.
- Suffers from occlusions.
- Requires a confined space to capture the motions limiting real world usability of the system.
- An older system. It is not using the latest technology available to increase performance.

Innalab 3DSuit

Advantages

- High precision orientation data.
- Not restrictive to a motion capture studio lab space.
- Easy setup and high subject comfort level.
- Multifunctional. Ability to connect 3 (Arm Tracker) to 30 sensors. Therefore, 3DSuit allows working with entire human body as well as with its limbs separately.
- Possibility of using 3DSuit in Animation Recording Mode and in Online Mode.
- Compatibility with optical motion capture systems.
- Compatibility with 3D animation packages - Motion Builder, 3DsMax, Maya.
- Minimal requirements to PC.
- Occlusion of body parts during active movements of actors has no influence on performance.

Disadvantages

- The Innalab 3DSuit has very poor documentation because the product is still in its beta stages of development.
- Heavy metals greatly interfere with the magnetometers, so large metal parts must be avoided. This limits the usability of the suit in a real world environment.
- Root position is calculated assuming a foot is always in contact with the ground. Therefore, vertical translation of the actor is captured incorrectly. Additionally the system cannot correctly capture a motion where both feet are removed from the ground.
- High translational errors (Figure B10) when using the foot sensors (i.e., motions that required walking). The foot sometimes moves backwards, and appears to slide during walking. Also, the error is very noticeable when the character walks in a circle and the ending position does not match the starting position. Because the translational error was roughly the same in multiple trials, we suspect it is just a result of Innalab not supplying the correct body measurements in the documentation.
- The foot sensor needed to be replaced part way through our session, suggesting the sensors are very fragile.

Figure B10: Inertial Suit Vertical Translation Error

Animazoo GypsyGyro-18

Advantages

- High precision orientation data.
- Does not require filters as inertial sensors inherently lack peak or noise.
- Rotations are taken directly from the actor's bones and then processed on suit meaning there are no complex software calculations resulting in data errors.
- Robust SDK (software development kit) and access to internal joint angle information.
- Automatic initial actor-calibration — simply 'drag and drop' points over an actor's digital photograph for faster and more accurate actor files.

Disadvantages

- Sensors are bulkier than optical, but can be worn under normal clothes.
- Root translation will need some cleanup.
- Still some magnetic field interference:
  Tolerance: 0.005 Gause@1m=1% Drift.
- Most Expensive of the inertial suits.
- Data does not reflect collection timestamps, ghost frames are added in the final capture and at times frames were dropped.
- Rigidity of the skeleton propagates errors to the effectors.
- Heavy metals greatly interfere with the magnetometers, so large metal parts must be avoided. This limits the usability of the suit in a real world environment.
- Tedious per capture calibration which is burdensome to the subject.
- No extended vertical translation capability.

Xsens Moven

Advantages

- High precision orientation data.
- The best accuracy of the 3 inertial systems tested.
- Automatic initial actor-calibration.
- Biometric skeleton for better accuracy.
- Robust SDK and access to internal joint angle information.

Disadvantages

- Still some magnetic field interference:
  Tolerance: 0.005 Gauss@1m=1% Drift.
- Direct contact with heavy metals is problematic.
- No extended vertical translation capability.